AMENDMENTS TO THE SPECIFICATION:

Please replace the paragraph starting at page 4, line 13 with the following:

Yet further features of the protocol of the present invention is the ability to provide read and write permissions to the information provided by the access to the server data, and the ability to modify the server state such as session[[.]], server and preferences.

Please replace the paragraph starting at page 5, line 3, with the following:

These and other objects, advantages and features of the present invention will become apparent to those skilled in the art from a study of the following Description of the Exemplary Preferred Embodiments when read in conjunction with the attached drawing and appended Claims.

Please replace the paragraph starting at page 5, line 12 with the following:

Referring now to Fig. 1, there is shown a computer network 10 constructed according to the principles of the present invention. Network 10 includes a server 12 and a client 14. the The communication between the server 12 and the client 14 may be either directly connected or through another medium 16, which may be any of a local area network, wide area network, the Internet, or any other proprietary or public network. Each of the server 12 and client 14 respectively have associated therewith a computer readable medium 18, 20, for storing of programs which implement the processes, procedures and methods of the herein below described inventive protocol.

Please replace the paragraph starting on page 10, line 7 with the following:

type= <- if defined then type checking of the server element type and the set type

is performed. If a match of the stored type and the request type fails, an error is returned

~~an~~ and the command fails.

Please replace the paragraph starting on page 11, line 17 with the following:

Data types can be any server allowed text value. New data types can be defined and

returned by the server. The data types are: Uint8, Sint8, Uint16, Sint16, Uint32, Sint32,

Uint64, Sint64, Float32, Float64, Bool8, Bool16, CharArray, QTSS_Object, and

void_pointer. QTSS_Objects, pointers and unknown data types always convert to a host

ordered hex value. The values converted are returned in hex. ~~AN~~ An example is when

adding a new attribute to the server. If the type is not known by the server, then the value

returned will be a string of hex values.

Please replace the paragraph starting on page 12, line 1 with the following:

A snapshot is the container of values from which an admin response is generated

(the measured values at a particular point in time, as compared to the constantly changing

values). There is one active snapshot per admin connection. Each request without a

snapshot in the URI generates a new snapshot. There is only one current snapshot per

request. Each new snapshot invalidates the previous ~~per~~ admin connection snapshot. A

snapshot is a hierarchy of containers specifying the type, name, value an read/write

attribute of values copied from the server's internal structures using the module API

(Application Program Interface).